

Tuomas Mäkipää

GE Fanuc 90-30-sarjan logiikoiden testausohjelma

Metropolia Ammattikorkeakoulu
Insinööri (AMK)
Automaatiotekniikka
Insinöörityö
24.05.2012

Tekijä	Tuomas Mäkipää
Otsikko	GE Fanuc 90-30-sarjan logiikoiden testausohjelma
Sivumäärä	29 sivua + 4 liitettä
Aika	24. touko 2012
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Automaatiotekniikka
Ohjaajat	Huoltoinsinööri Antti Kuusela Lehtori Markus Rimpivaara
<p>Tämän työn tarkoituksena oli luoda Navis Engineering Oy:lle GE Fanuc 90-30-sarjan ohjelmoitavien logiikoiden testausohjelmia. Testausohjelmilla pystytään testaamaan toiminnaltaan epävarmat prosessorimoduulit sekä IO-moduulit, jotta ne voidaan ottaa käyttöön ilman pelkoa siitä, että ne ovat rikki tai vikaantuvat heti.</p> <p>Ohjelmat tehtiin GE Intelligent Platformsin Proficy Machine Edition -ohjelmalla, joka on GE Fanucin VersaPro -ohjelmointiohjelman seuraaja. Tämä työ käy läpi miten prosessorin ja sen muistin sekä digitaalisten IO:iden testaus tapahtuu ja miten testejä voi soveltaa muiden moduulien testaamiseen.</p>	
Avainsanat	Fanuc, 90-30, testaus

Author	Tuomas Mäkipää
Title	A Test Program for the GE Fanuc 90-30 Series PLC
Number of Pages	29 pages + 4 appedices
Date	22 May 2012
Degree	Bachelor of Engineering
Degree Programme	Automation Tehchnology
Instructors	Antti Kuusela, Service Engineer Markus Rimpivaara, Senior Lecturer
<p>The purpose of this study was to create a test program for GE Fanuc Series 90-30 programmable logic controllers. Test programs can ensure functionality of operationally uncertain processor modules and IO modules, so that they can be put to use without fear of them being broken or failing.</p> <p>The programs were made by using the GE Intelligent Platforms' Proficy Machine Edition software, which is a successor of the GE Fanucs VersaPro programming software. This thesis introduces how to test CPU memory, and digital IO modules and how the test can be applied to other modules.</p>	
Keywords	Fanuc, 90-30, test

Sisälllys

1 Johdanto	1
2 Navis Engineering Oy	1
2.1 Yritys lyhyesti	1
2.2 NavDP 4000	2
2.3 Logiikat NavDP 4000 -järjestelmässä	2
2.4 Testiohjelman tarve	2
3 Ohjelmoitavat logiikat	3
3.1 Yleistä logiikoista	3
3.2 Logiikan osat ja toiminta	3
3.3 GE Fanuc 90-30 -sarjan logiikat	4
3.3.1 Virtalähde	5
3.3.2 Backplane	5
3.3.3 Prosessori	6
4 Proficy Machine Edition	7
4.1 Yleistä	7
4.2 Yhteysasetukset	7
4.3 Hardware-asetukset	8
4.4 Logiikkaohjelman lataus PLC:hen	9
4.5 Help	9
5 Ohjelmointi ja tiedonsiirtoasetukset	10
6 Prosessori ja muisti	12
6.1 Muistin kuvaus	12
6.2 Ohjelman kuvaus	14
6.3 Ohjelman rakentaminen	14
6.4 Ohjelman kulku	15
6.4.1 Aloitus	15

6.4.2	Bittien kirjoitus muistiin	15
6.4.3	Bittien luku ja tarkistus	17
6.4.4	Muistin resetointi	19
7	Testaus	19
7.1	Testin kuvaus	19
7.1	Testin vaiheet	20
7.2	Ongelmatilanteet testin aikana	21
8	Digitaaliset IO-moduulit	22
8.1	IC693MDR390 Mixed-IO -testi	22
8.1.1	Laitteen kuvaus	22
8.1.2	Hardware-asetukset	23
8.1.3	Testiohjelma	25
8.1.4	Kytkenät	26
8.1.5	Testaus	27
9	Päätelmät	28
	Lähteet	29
	Liitteet	
	Liite 1. NavDP 4000 -kaavio	
	Liite 2. Moduulien asennusohje	
	Liite 3. BIT_SET_WORD -lohkon esimerkki	
	Liite 4. Järjestelmän tilabittejä	

Lyhenteet ja käsitteet

AC	<i>Altering Current.</i> Vaihtovirta.
Bitti	Bitti on pienin mahdollinen tiedon tallennusmuoto. Bitit ovat ykkösiä ja nollia.
BSW	<i>Bit_set_word</i> -lohko. Ohjelmointilohko, jolla kirjoitetaan bittejä logiikan muistiin.
BTW	<i>Bit_test_word</i> -lohko. Ohjelmointilohko, jolla luetaan bittejä logiikan muistista.
CPU	<i>Central processing unit.</i> Prosessori.
DC	<i>Direct Current.</i> Tasavirta.
IO	<i>Input/Output.</i> Tulot ja lähdöt.
PLC	<i>Programmable logic controller.</i> Ohjelmoitava logiikka.
PME	<i>Proficy Machine Edition.</i> GE Intelligensin ylläpitämä ohjelmointiohjelmisto.
PROM	<i>Program ROM.</i> Muisti, josta voi ladata ohjelman logiikan käynnistyessä.
RAM	<i>Random Access Memory.</i> Käyttömuisti.
ROM	<i>Read Only Memory.</i> Lukumuisti.
SCU	<i>Serial Communications Utility.</i> PME:n tiedonsiirron työkalu.
Word	Word on 16 bitin mittainen tiedon tallennusmuoto.

1 Johdanto

Tämän työn tarkoituksena on luoda Navis Engineering Oy:lle GE Fanuc 90-30-sarjan logiikkojen moduulien testausohjelmia. Moduulien testausta tarvitaan erityisesti silloin, kun GE Fanucin 90-30-sarjan tuotanto lakkautetaan. Tuotannon loputtua sarjan komponenttien hinta nousee, koska varaosia ei ole enää saatavilla ja varastoon kertyneet ylimääräiset moduulit otetaan käyttöön.

2 Navis Engineering Oy

2.1 Yritys lyhyesti

Navis Engineering Oy (lyhyemmin Navis) on vuonna 1992 perustettu dynaamisen paikallistamisen kehittäjä ja järjestelmätoimittaja. Yritys tähtää valmistamaan ja toimittamaan korkealaatuisia, turvallisia ja luotettavia laivojen automaatio- ja paikannusjärjestelmiä meriteollisuudelle. Navisin tuotteisiin kuuluu dynaamisen paikannusjärjestelmän (NavDP-4000) lisäksi laivojen ohjainjärjestelmiä sekä autopilot-järjestelmiä.



Kuva 1. JP-4000 -ohjausjärjestelmän käyttöpaneeli

Navis Internationalin pääkonttori sijaitsee Vantaalla, missä järjestelmät myös kasataan ja testataan ennen kohteeseen vientiä. Navisin suunnittelu ja kehitys tapahtuu Pietaris-
sa. Navis myös ylläpitää palvelu-, teknologia- ja jakelukumppanien kasvavaa verkkoa ympäri maailmaa. Yritys työllistää yli 50 henkilöä eri aloilta. [1]

2.2 NavDP 4000

NavDP 4000 on moderni laivoihin tarkoitettu paikannus- ja ohjausjärjestelmä, joka asennetaan laivoihin sellaisenaan tai integroidaan valmiiseen järjestelmään. NavDP 4000:n tarkoitus on taata laivan vakaa ohjaus sekä paikoillaan pysyvyys tarvittaessa silloin, kun ihminen ei siihen pysty. Järjestelmässä toimii aivoina laivasta luotu matemaattinen malli, johon ohjaus perustuu.

2.3 Logiikat NavDP 4000 -järjestelmässä

Matemaattinen mallinnus tapahtuu tietokoneella. Tietokone saa laivan antureista sekä ohjaimista tiedot mallinnusta varten. Mallinnus luo voimavektorit thrustereille ja peräsimille, joita ohjataan PLC:illä.

Järjestelmässän melkein kaikki on PLC:llä ohjattua. Vain antureiden tiedot luetaan suoraan mallinnuskoneilla. Yleensä DP järjestelmän komponentit ovat monistettuja. Jos yksi vikaantuu, niin toisella voi vielä ohjata. PLC:llä ohjataan NavDP-"järjestelmiä" päälle ja pois tarpeen mukaan. Järjestelmän ohjaustiedot lähtevät ohjauspaneelin PLC:n kautta thrustereita ja peräsimiä ohjaaville PLC:ille. Järjestelmän kaaviokuva liitteenä 1.

2.4 Testiohjelman tarve

Testausohjelman avulla voidaan todeta epäluotettavien moduulien toimivuus. Myös "virheellisten valmistuserien" moduulien seassa on suurin osa toimivia yksiköitä, jotka eivät ole valmistuseränsä vuoksi tarpeeksi luotettavia otettavaksi käyttöön sellaisenaan. Ohjelman avulla voidaan varmistaa moduulien toimivuus, minkä jälkeen moduulit voidaan käyttää tai myydä varaosiksi. Testaamisesta on hyötyä eritoten silloin, kun 90-30-sarjan tuotanto lopetetaan ja varaosista alkaa olla pulaa.

Tässä työssä perehdytään digitaalisten IO-moduulien lähtö- ja sisääntulojen, muistin sekä prosessorin toiminnan tarkistamiseen ohjelmallisesti. IO-moduuleissa jokainen IO testataan. Muistin testauksessa käydään muistipaikkojen toimivuus läpi bitti kerrallaan.

3 Ohjelmoitavat logiikat

3.1 Yleistä logiikoista

Ohjelmoitavat logiikat ovat pieniä digitaalisia tietokoneita, joilla ohjataan automaattisia laitteita ja muita sähköisiä prosesseja. Yleisiä käyttökohteita ovat esimerkiksi tehtaiden tuotantolinjastot, moottorihjaukset ja rakennusautomaatio, kuten ilmastoinnin ohjaus. Logiikat saivat alkunsa Yhdysvaltain autoteollisuudessa, josta ne ovat levinneet muihin käyttösovelluksiin. [2]

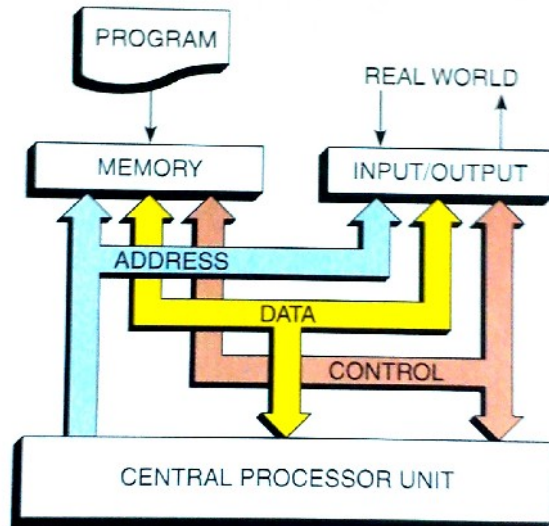
Logiikan ohjelmointi tapahtuu tietokoneeseen asennettavalla ohjelmointiohjelmistolla. Ohjelmistot ovat yleensä logiikan valmistajien itse tekemät ja ne kelpaavat vain valmistajien omien logiikkojen ohjelmoimiseen. Ohjelmointiohjelmalla ohjelmoidut ohjelmat siirretään logiikkaan Ethernet-, RS323-, RS422- tai RS485, -kaapeleilla.

Ensimmäiset logiikat ohjelmoitiin ladder diagram- sekä instruction list -kielillä, jotka ovat käytössä vielä nykyäänkin. Sittenmin on standardisoitu useampia ohjelmointikieliä, jotka ovat FBD (Function Block Diagram), LD (Ladder Diagram), ST (Structure Text), IL (Instruction List) ja SFC (Sequential Function Chart). [2]

Tunnetuimpia logiikkojen valmistajia suomessa GE Fanucin lisäksi ovat esimerkiksi Siemens, Beckhoff ja Mitsubishi. Tässä insinööriyössä keskitytään GE Fanucin 90-30-sarjan logiikoihin ja niiden testaukseen.

3.2 Logiikan osat ja toiminta

Logiikat koostuvat useimmiten neljästä komponentista: prosessorista, virtalähteestä, sisääntuloista ja ulostuloista. Nämä komponentit ovat usein modulaarisia, eli toisistaan erillisiä osia. Joissain prosessoreissa on valmiiksi rakennettuna sisään- ja ulostuloja. Niitä kutsutaan kiinteiksi IO:ksi (fixed io).



Kuva 2. Logiikan osiot ja niiden vuorovaikutus [3, s.7]

3.3 GE Fanuc 90-30-sarjan logiikat

90-30-sarjan logiikat koostuvat useimmissa tapauksissa backplanesta, virtalähteestä, prosessorista sekä muista lisämoduuleista. Joissain tapauksissa CPU on integroitu backplaneen.



Kuva 3. GE Fanuc -logiikka ja sen moduulit koottuna

3.3.1 Virtalähde

Virtalähde kytketään 100-240 V AC-virtalähteeseen tai 125 V DC-lähteeseen, jolloin se antaa logiikalle sen tarvitseman sähkövirran. Virtalähteen kannessa on pääjännitteen liittimien lisäksi myös 24 voltin liittimet, joista saa tarvittaessa 24 V jännitettä ulos.



Kuva 4. Virtalähde

Virtalähteen kannesta löytyy neljä valoa. Valot ovat PWR, OK, RUN ja pariston jännitteen valoja. PWR ilmaisee jännitelähteeseen tulevan jännitteen, OK ilmaisee, että logiikan kokoonpanossa eikä asetuksissa ole virheitä (ei ole rikki, löytää backplanen ja prosessorin) ja RUN ilmaisee, onko logiikkaohjelma päällä vai pois. Patterin valo ilmaisee patterin jännitteen olevan alhainen. OK- ja RUN -valot ovat vihreitä ja patterin valo on punainen.

Virtalähteessä on paikka patterille, joka ylläpitää RAM-muistia, kun logiikka ei saa virtaa verkosta. Muisti nollaantuu ja logiikkaohjelma ja sen muuttujat katoavat, kun jännite loppuu.

3.3.2 Backplane

Backplane on virtalähteen, prosessorin ja moduulien alusta. Kaikki moduulit kiinnitetään backplaneen, jonka välityksellä ne siirtävät tietoa toisilleen. Backplaneen saa kiinnitettyä vain rajatun määrän moduuleita, joten suurempia logiikkajärjestelmiä varten täytyy olla erillisiä alustoja lisämoduuleille, tai yksinkertaisesti pitää käyttää useampaa PLC:tä



Kuva 5. Backplane, jossa on prosessori integroituna

3.3.3 Prosessori

Prosessorit toimivat logiikkakontrollerien aivoina. Ne vastaanottavat, käsittelevät, lähettävät ja varastoivat kaiken tiedon, mitä logiikassa liikkuu. Prosessorin kannesta voi kääntää prosessorin on- ja off -tilaan avaimella. Joskus prosessori on integroitu backplaneen.



Kuva 6. IC693CPU374 -prosessorimoduuli

Logiikan muisti sijaitsee prosessorissa. Muistia on kahden tyyppistä. RAM- ja PROM-muistit. PROM -muisti toimii ohjelman backup-muistina, eikä se tyhjene, vaikka PLC:stä katkeaisi virta. RAM-muisti toimii käyttömuistina ohjelmia käytettäessä ja se tyhjenee virran katketessa. PROM-muistiin tallennetaan tarvittaessa ohjelmat, josta ne luetaan RAM-muistiin aina, kun PLC käynnistyy. Tällöin ohjelma ei katoa sähkökatkojen tai muun virransyöttöhäiriön seurauksena. PROM-muistiin ei voi kirjoittaa PLC:n ollessa RUN-tilassa.

4 Proficy Machine Edition

4.1 Yleistä

GE Intelligent Platformsin Proficy Machine Edition on universaali kehitysympäristö teollisuuden käyttöliittymille, liikkeenohjaus- ja ohjausjärjestelmille. Proficy Machine Edition tarjoaa yhteisen käyttöliittymän kontrollereiden ohjelmointiin, käyttöliittymien tekoon sekä moottorinohjauksiin. [4]

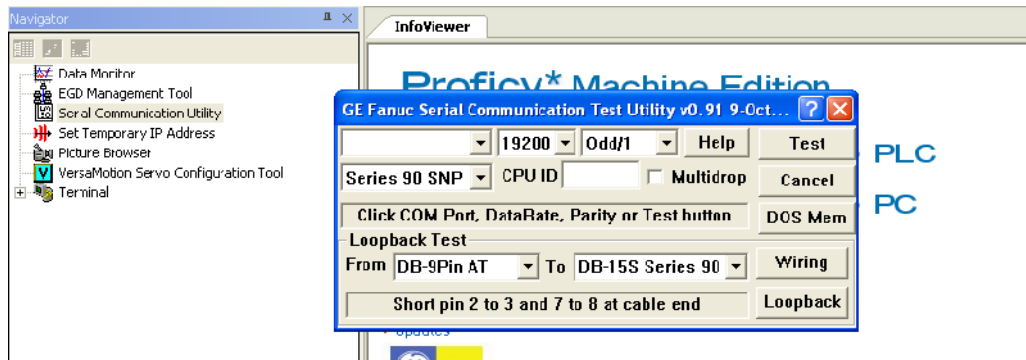
PME sisältää neljä komponenttia, View, Logic Developer PC, Logic Developer PLC ja Motion Developer. View'llä voi tehdä käyttöliittymiä, Logic Developer PC:llä tehdään PC-pohjaiset ohjaukset, Logic Developer PLC:llä tehdään ohjaus ohjelmoitavaan logiikkaan ja Motion Developerilla tehdään moottoreiden ohjauksia. [4]

PME:llä ohjelmoidaan PACSystems-, 90-30-, 90-70 -sarjojen logiikoita sekä etä-IO-moduuleita.[4]


4.2 Yhteysasetukset


PME:ssä on oma Serial Communications Utility -työkalu, jolla voidaan selvittää logiikan yhteysasetukset vaivatta. SCU:n löytää PME:n Navigator-ikkunasta utilities-välilehdeltä. Työkalu käy läpi tietokoneen COM-portit ja etsii sen portin, johon logiikka on kiinnitetty. Se etsii myös yhteyden nopeuden ja muut asetukset. Logiikassa pitää olla virta päällä kun SCU-työkalua käytetään. SCU ei aseta löytämiään asetuksia käytettäväksi, vaan ne


pitää asettaa Windowsin Device Managerista ja logiikan CPU-asetuksista.




Kuva 7. PME:n Navigator-ikkuna ja SCU-työkalu

Yhteyden saamiseksi täytyy valita oikea COM-portti, johon ohjelmointikaapeli on kytketty. COM-portin voi valita valitsemalla Navigator-ikkunasta käynnissä olevan projektin "Targetin". Target  tarkoittaa laitetta, johon otetaan yhteys ja jonka nimi on oletuksena Target. Targetin ollessa aktiivinen voi sen asetuksia tarkastella Inspector-ikkunasta. COM-portti löytyy Physical Port -kohdasta.

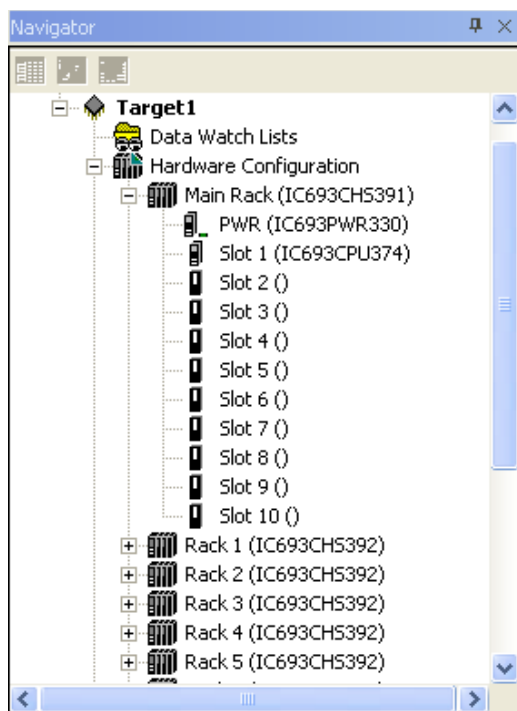
PME muodostaa yhteyden logiikkaan painamalla työkalupalkissa olevaa salaman kuvaa . Vaihtoehtoisesti työkalupalkin Target-valikosta voi valita "Go Online with Target X".

Saatuun yhteyden logiikkaan PME on online-tilassa. Online-tilan saa pois työkalupalkin Toggle Online mode -painikkeesta . Tiloja on kaksi, online ja offline tai toisin sanoen Monitor- ja Developer-tilat. Monitor-tilassa voi vain tarkastella ohjelmaa ja developer-tilassa ohjelmaan voi tehdä muutoksia.

4.3 Hardware-asetukset


Logiikan hardware-asetukset löytyvät PME:n Navigator-ikkunan Project-välilehdeltä. Project-välilehdellä on oletuksena target-valikko . Target viittaa yksittäiseen logiikkaan. Ohjelmassa voi olla useampia target-logiikoita. Targetin alta löytyy hardware configuration -valikko jossa hardware-asetukset määritellään.

Logiikan moduulit voi määrittää painamalla hiiren oikeaa näppäintä moduulin slotin kohdalla ja valitsemalla add/replace module. Logiikan CPU-sekä kaapeliyhteys-asetukset löytyvät CPU-slotin kohdalta hiiren oikealla klikkauksella aukeavasta valikosta valitsemalla configure.



Kuva 8. Navigator-ikkunan hardware-osio

4.4 Logiikkaohjelman lataus PLC:hen

Ohjelma voidaan ladata logiikalle, kun siihen on saatu yhteys. PME:n pitää olla offline-tilassa. Ohjelma ladataan PLC:hen painamalla "download to controller" -nappeja  tai painamalla F8:aa.

4.5 Help

PME:ssä löytyy lisäapua melkein mistä tahansa osasta hiiren oikealla klikkauksella. Logiikkaohjelmaan vinkkejä ja apua löytyy koko ajan Companion-ikkunasta, jossa on tiivistettynä aktiivisena olevien osien (klikatut osat) tiedot. Companion on oletusasetuksilla aina näkyvissä. Syvemmällistä tietoa on kuitenkin usein tarpeellista hakea ihan oikeasta helpistä. F1-näppäin avaa klikattujen kohteiden infisivun Infoviewer-ikkunaan.

5 Ohjelmointi ja tiedonsiirtoasetukset

Jokaisen GE Fanuc 90-30-sarjan prosessorin tai virtalähteen kannessa on 15-pinninen RS-422 yhteensopiva portti. Portin kautta logiikkaohjelma ladataan logiikkaan ja tarkkailla logiikan muuttujia.



Kuva 9. GE Fanuc -virtalähde ja sarjaportti

Tässä työssä yhteys logiikkaan otetaan IC690USB901 USB to RS-422 SNP -kaapelilla.

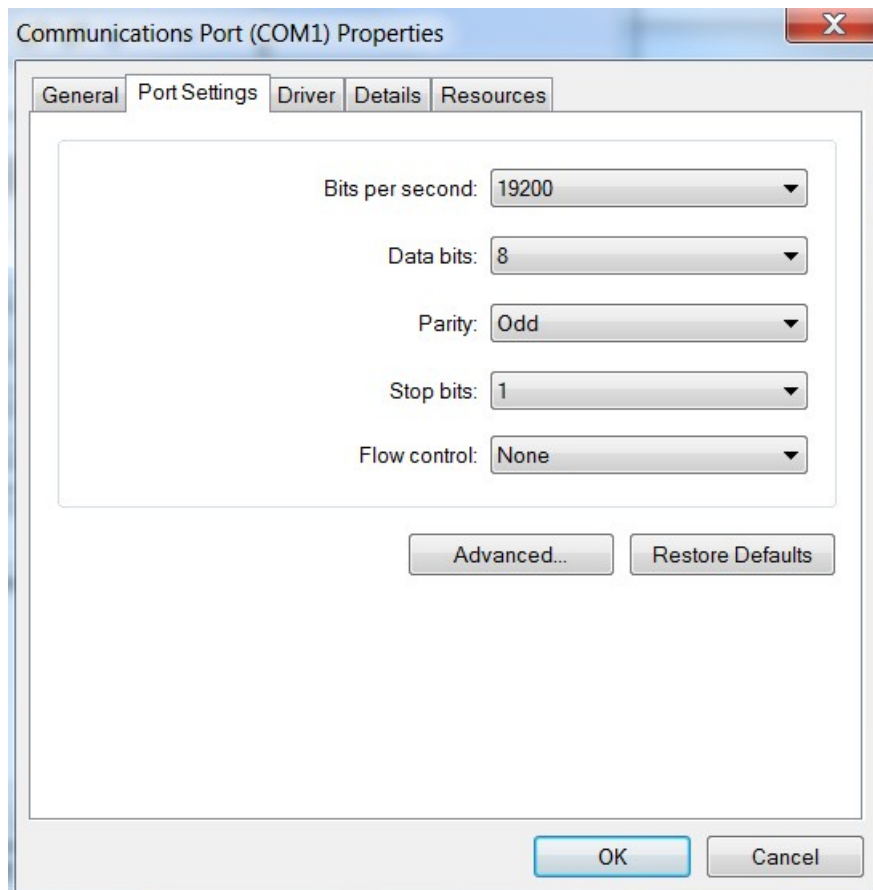


Kuva 10. IC690USB901 -ohjelmointikaapeli

Tietokoneen ja PLC:n yhteyttä varten täytyy tietokoneeseen asentaa ajurit ohjelmointikaapelille. Myös yhteyden asetukset täytyy määrittää erikseen tietokoneelta ja PLC:ltä. GE Fanucin oletusasetukset ovat seuraavat [5]:

- Databits: 8
- Stop bits: 1
- Parity: Odd
- Baud rate: 19 200 bps

Kaapelin asetuksiin pääsee käsiksi tietokoneen laitehallinnasta (Start -> Control Panel -> Device Manager). Laitehallinnasta etsitään Ports-välilehti ja sieltä COM-portti, johon kaapeli on kytketty. Portin asetuksiin pääsee käsiksi hiiren oikealla näppäimellä ja valitsemalla Properties. Propertiesista löytyy Port settings -välilehti, josta asetukset löytyvät.



Kuva 11. COM-portin asetukset device managerissa

PLC:n puolelta asetukset löytyvät ohjelmointiohjelmalla. Proficy Machine editionilla asetuksia käydään läpi luvuissa 3.1 ja 3.2.

6 Prosessori ja muisti

6.1 Muistin kuvaus

Ge Fanuc -logiikoiden muistit ovat integroituina prosessorimoduuleihin, joten muistia testatessa tarkastellaan myös prosessorimoduulia. Tässä työssä oletetaan, että prosessori on toimiva, jos muistitesti toimii ongelmitta.

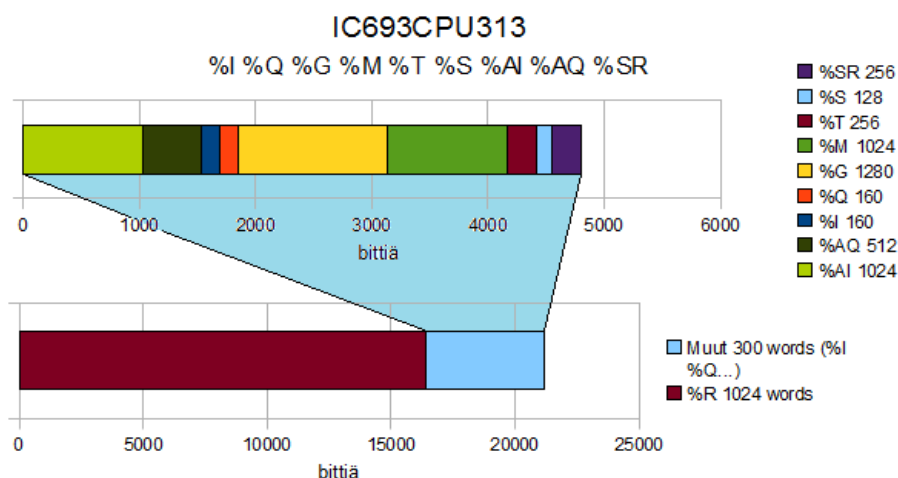
GE Fanuc -prosessorien muisti koostuu monesta erikokoisesta osiosta. Kaiken kaikkiaan osioita on yli 10. Joitain muistipaikkoja on varattu PLC:n omiin toimintoihin, joihin ei pääse käsiksi sekä logiikkaohjelman toiminnoille. Muistista pyritään kuormittamaan vain

muistipaikkoja %I, %Q, %M, %T, %R, %AI, %AQ ja %G, koska niihin pääsee kirjoittamaan logiikkaohjelman avulla. Muistialueista %S, %SA, %SB, %SC eli system muisteista tarkistetaan mahdolliset virheilmoitukset ohjelman alussa. %S-, %SA-, %SB- ja %SC -muisteissa olevat muuttujat ovat logiikan käytössä risuaita-alkuisilla muuttujilla, esimerkiksi #FST_SCN ilmaisee ensimmäistä ohjelmakierrosta ja sen muistipaikka on %S0001. %S muistiin ei voi kirjoittaa, mutta %SA %SB %SC muisteihin voi. [6]

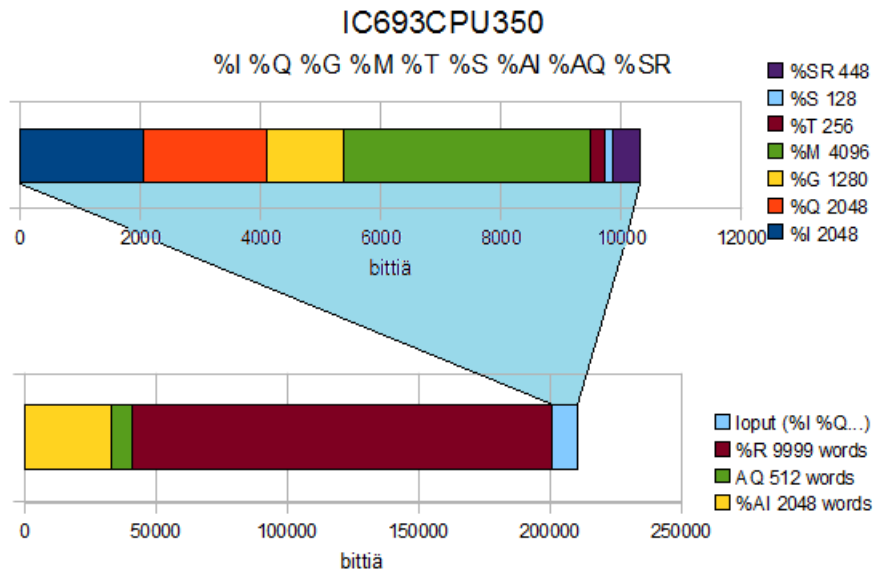
Taulukko 1: Muistin osiot

%I	Diskreetit sisääntulojen pisteet
%Q	Diskreetit ulostulojen pisteet
%M	Sisäiset muuttujat
%T	Väliaikaismuuttujat (ulostulo)
%R	Rekisterimuisti
%AI	Analogiset sisääntulot
%AQ	Analogiset ulostulot
%G	Diskreetti globaali muisti
%S	Järjestelmän tilat

Eri prosessorimalleissa on erikokoiset muistit. Jokainen prosessorimalli tarvitsee siis uniikin ohjelman muistin testaamiseen. Tässä työssä panostetaan siis esimerkein ohjelmien avainkohtiin, joiden avulla voidaan rakentaa muistitesti jokaiselle prosessorimallille erikseen.



Kuva 12. IC693CPU313 -muistialueet [7]



Kuva 13. IC693CPU350 muistialueet [7]

6.2 Ohjelman kuvaus

Ohjelma kirjoittaa logiikan muistipaikkojen %I, %Q, %M, %T, %R, %AI, %AQ ja %G bitit ykkösiksi. Tarkoituksena on kokeilla jokaisen muistipaikan kohdalla, että muistiin voi kirjoittaa ja sieltä voi lukea.

Bittien ollessa päällä ohjelma tarkistaa bitti kerrallaan, että ne ovat päällä. Samalla muisti kuormittuu normaalia enemmän, koska normaalia suurempi määrä muistia on käytössä samanaikaisesti.

Proessorit, joissa on vialliset muistit, eivät todennäköisesti edes mene OK-tilaan (virtalähteen OK-valo päälle).

6.3 Ohjelman rakentaminen

Ohjelman rakentamiseksi tarvitaan ohjelmointiohjelma. Tässä työssä ohjelmointi tapahtuu Proficy Machine Edition Logic Developer PLC:llä. Ohjelmointiin voi myös käyttää PME:n edeltäjää VersaProta.

Muistitestin esimerkkinä käydään läpi IC693CPU313-mallille rakennetun muistin tes-

tausohjelman pääkohtia. IC693CPU313 on integroitu backplaneen ja se sisältää vähemmän muistia kun muut prosessorimallit, mutta muistin toimintaperiaate on kaikissa prosessoreissa sama.

Ohjelmaan rakennetaan neljä osiota. Main-osiossa pyöritetään koko ohjelmaa. Se ohjaa testausprosessin vaiheita, joita ovat bwrite, bcheck ja reset.

6.4 Ohjelman kulku

6.4.1 Aloitus

Kun logiikka menee run-tilaan, se aloittaa ohjelman suorittamisen main-lohkosta. Ensimmäiseksi on hyvä tutkia PLC:n oman diagnostiikan virheilmoituksia #GRD_FLT, #SFT_FLT, #BAD_RAM. Jos jokin näistä muuttujista on tosi, ohjelman ei kannata edetä, ennen kuin tiedetään, missä vika on. Tällöin tarkastetaan fault-aulun virheilmoitukset.

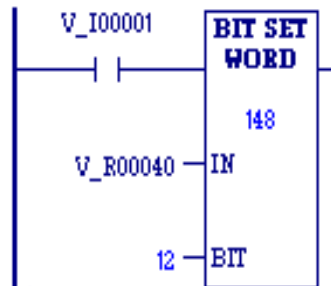
Fault-auluun ilmestyvät virheilmoitukset saattavat asettaa logiikan stop-tilaan. PLC-asetuksista voidaan kuitenkin asettaa ohjelma olemaan välittämättä virheistä, jolloin plc pysyy run-tilassa, vaikka virheitä löytyisikin.

Tämä aloitusvirhetesti ei liity mitenkään muistin testaamiseen, vaan sen on tarkoitus kiinnittää huomio logiikan oman virhediagnostiikan virheilmoituksiin. PLC:tä testatessa pitää aina muistaa tarkistaa ja tyhjentää fault-aulu. Laitteita on turha testata käyttöön palautettavaksi, jos laitteiston oma diagnostiikka huomaa olevansa vikaantunut eikä virhettä saa korjattua. Tällöin moduulit kelpaavat korkeintaan ohjelmoijien leikkikaluiksi.

6.4.2 Bittien kirjoitus muistiin

Muistipaikkojen kirjoitus tapahtuu bwrite-lohkosta käsin. Jokaiselle muistityypille %I, %Q, %M, %T, %R, %AI, %AQ ja %G on oma bittejä kirjoittava BIT_SET_WORD -lohkonsa. BIT_SET_WORD-lohkoon viitataan tästä lähin lyhenteellä BSW. %R-muistille BSW-lohkoja on useampi, koska %R-muistin koko on 1024 wordia ja BSW-lohko voi kir-

joittaa vain 256 wordin pituisille muistialueille. IC693CPU313:n koko muistiin kirjoittamista varten BSW-lohkoja tarvitaan siis 11 kappaletta.



Kuva 14. BSW-lohko

Kuvan 14 lohko kirjoittaa ykkösen 12 bitin päähän V_R00040-muuttujan ensimmäisestä bitistä. Tällä loholla voi muuttaa bittejä ykkösiksi 148 wordin, eli $148 \times 16 = 2368$ bitin pituudelta V_R00040-muuttujan ensimmäisestä bitistä. Esimerkki BSW-lohkon toiminnasta on liitteessä 3.

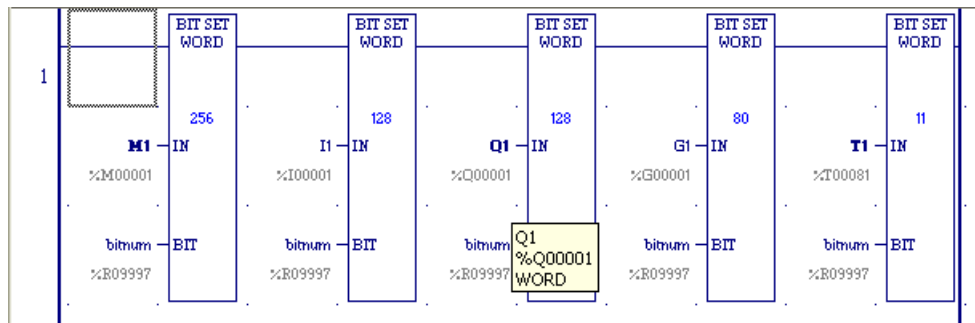
Kun BSW-lohko saa signaalin, se lukee IN- ja BIT-sisääntuloista tiedot. IN sisältää osoitteen, mihin muistin kohtaan ensimmäinen kirjoitettava bitti tulee. BIT-sisääntulo kertoo, kuinka monen bitin päässä vaihdettava bitti sijaitsee IN:n osoitteesta. IN-osoitteena voi toimia word-muuttuja. Muut muuttujatyypitkin toimivat, mutta ne aiheuttavat varoitusilmoituksen.

BSW-lohkot eivät kuitenkaan anna kirjoittaa bittejä kuin maksimissaan 256 wordin pätkissä. Kirjoitettavien wordien määrä voidaan valita väliltä 1-256. 1024 wordin kokoisen %R-rekisterin täyttämiseen tarvitaan 4 kappaletta BSW-lohkoja. BIT- sisääntuloon tuleva bitnum -muuttuja kasvaa siis enimmillään $256 \times 16 = 4096$ bittiin, jonka jälkeen bitnum -laskuri resetoituu.

Yksi BSW-lohko täytyy asettaa kirjoittamaan vajaan määrä wordeja, jotta saadaan varattua muistipaikat tarvittaville laskureille ja muille word-rekisteriä tarvitseville muuttujille. Sama sääntö pätee myös muihin muistialueisiin.

Kun BIT-sisääntulon numero kasvaa yli BSW-blokissa määritellyn word-alueen, BSW-lohko ei päästä signaalia lävitseen. Tämä kannattaa muistaa, jos BSW-lohkoja sijoitte-

taan peräkkäin ohjelmassa. Tällöin lohkot sijoitetaan siten, että suurin alue on ensin.



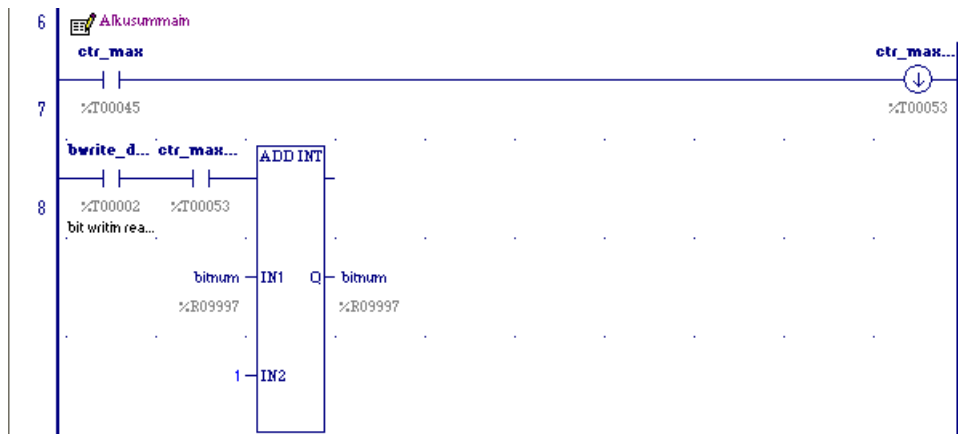
Kuva 15. BSW-lohkot peräkkäin IC693CPU350-testiohjelmassa

6.4.3 Bittien luku ja tarkistus

Kun bitnum on ensimmäisen kerran päässyt arvoon 4096, bitnum -laskuri resetoituu, bwrite-lohkossa bwrite_done-muuttuja vaihtuu todeksi ja katkaisee bwrite-lohkon kutsusignaalin main-lohkossa. Bcheck-lohko saa signaalin bwrite-osion loputtua ja bwrite_done-muuttujan muututtua todeksi. Bittien tarkistus voi alkaa.

Bittien tarkistus tapahtuu periaatteessa samaan tapaan kuin bittien kirjoitus. Bit_test_word-lohko lukee muistialuetta samaan tapaan kuin BSW-lohko kirjoittaa. Bit_test_word-lohkoa kutsutaan tästä eteenpäin BTW-lohkoksi.

Erona BSW:hen BTW-lohko tarvitsee bitnumin arvoksi 1, jotta se voi lukea ensimmäisen bitin. Nollasta ei voi aloittaa. Sitä varten on tehty bitnum-laskuria resetoivan muuttujan ja bwrite_done-muuttujan avulla alkusummaaaja. Alkusummain on yksinkertaisesti summalohko, joka nostaa bitnumin arvon nolasta yhteen bittien lukemisen alkaessa. Kun bitnum on taas 1, BTW-lohkot osaavat lukea muistia oikeasta kohdasta.



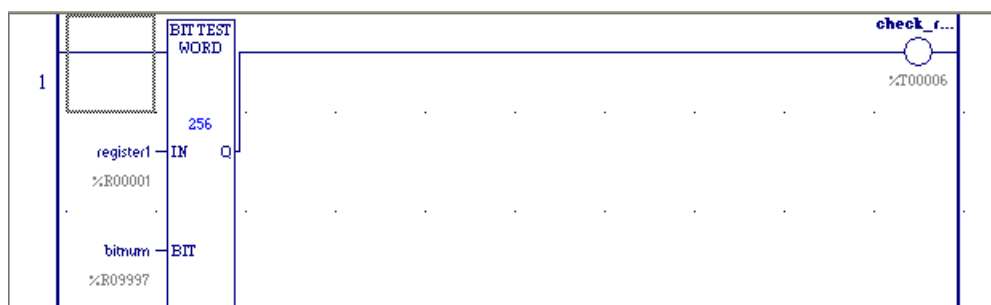
Kuva 16. Alkusummain

BTW-lohkoissa on sisääntulojen lisäksi ulostulo Q, joka indikoi luetun bitin tilaa. Kun luettu bitti on tosi, on Q:n arvo myös tosi. BTW-lohkojen ulostuloja seuraa check01 - check11-muuttujat.



Kuva 17. Bit_test_word-lohko

BTW-lohkojen check01...11 muuttujat ilmaisevat lohkojen lukeman muistin tilaa. Ohjelma ei jatka bittien lukua, elleivät kaikki check01...11-muuttujat ole tosia. Näin ohjelman pysähtyessä nähdään, mikä muistin bitti on pois päältä, ja voidaan paikallistaa missä muisti reistailee.

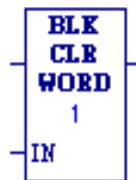


Kuva 18. BTW-lohko ohjelmassa.

Ohjelma jatkaa check01...11:n tarkistamista ja bitnumin kasvattamista, kunnes kaikki muistipaikat ovat testattu eli bitnum-muuttuja on kasvanut arvoon 4096. 4096:n jälkeen bitnum resetoituu ja ohjelma siirtyy resetoimaan kirjoitetut bitit nolliksi.

6.4.4 Muistin resetointi

Muistin resetointi tapahtuu bittien tarkistuksen jälkeen. Muisti tyhjennetään BLK_CLEAR_WORD-lohkolla, joka kirjoittaa muistin kaikki bitit nolliksi.



Kuva 19. BLK_CLEAR_WORD-lohko

BLK_CLEAR_WORD-lohko toimii samalla periaatteella kuin BSW- ja BTW-lohkot. Kun lohko saa virtaa, se resetoi kaikki muistipaikat sen alueella, joka on 1-256 wordia. IN-sisääntuloon tulee ensimmäisen resetoitavan bitin osoite.

7 Testaus

7.1 Testin kuvaus

Tämän insinööriyön testitavalla voidaan 90-30-sarjan prosessorit muisteineen todeta toimiviksi. Pyörittämällä testiä pidemmän aikaa voidaan päätellä, että prosessori kestää pidempiaikaista kuormitusta.

Testi ei kokeile integroitujen CPU:iden alustan moduulipaikkojen toimintaa. Moduulien tai moduulipaikan rikkinäisyys täytyy todeta asentamalla moduulit kiinni ja ajamalla

ohjelma, joka käyttää niitä. Tällöin moduulien rikkonaisuusilmoitus ilmestyy fault-tauluihin.

7.2 Testin vaiheet

Seuraavia vaiheita seuraamalla voidaan tarkistaa prosessorin toiminta sekä RAM ja PROM-muistien toiminta. Nämä vaiheet soveltuvat hyvin muistisäännöksi muidenkin ohjelmien lataamiseen PLC:hen seuraavassa järjestyksessä.

1. Kiinnitetään tarvittavat moduulit ja lisäosat laitteistoon.
2. Kytetään PLC:hen virta.
 - OK-valo syttyy.
3. Kiinnitetään ohjelmointikaapeli.
4. Muodostetaan yhteys logiikkaan ohjelmointiohjelmalla.
5. Tarkistetaan CPU-asetukset
 - Asetetaan CPU lataamaan ohjelma ja rekisterit PROM muistista.
 - Asetetaan logiikka menemään run tilaan käynnistyessä.
6. Ladataan ohjelma logiikan RAM-muistiin ohjelmointiohjelman download to controller -napista tai painamalla näppäimistön F8:aa.
7. Ladataan ohjelma PROM-muistiin ohjelmointiohjelman valikosta target -> online commands -> flash/EEPROM. (Prossessori avaimesta off-tilassa).
8. Katkaistaan yhteys logiikkaan.
9. Katkaistaan logiikasta virta.
10. Kytetään logiikkaan virta takaisin.
 - Logiikka menee run-tilaan.
11. Otetaan yhteys logiikkaan ja seurataan testin etenemistä, tai annetaan logiikan pyörittää testiä itseksensä.
12. Kun ohjelma on pyörinyt tarpeeksi pitkään ilman ongelmia, voidaan prosessori sekä RAM- ja PROM-muistit todeta toimiviksi.

13. Asetetaan logiikka stop-tilaan. Tyhjennetään muisti PME:n reset-napilla.
14. Ladataan logiikkaan CPU:lle asetukset (kohta 5), joilla se lukee RAM-muistia PROMin sijaan käynnistyessään.

Jos logiikka yrittää lukea PROMMista tyhjää tai PROM-muistipalikka on otettu irti (IC693CPU313), se ei välttämättä pääse OK-tilaan (ok-valo ei syty käynnistyessä), jolloin yhteyden saanti logiikkaan voi olla hankalaa ja koko järjestelmä vaikuttaa rikkiinäiseltä.

7.3 Ongelmatilanteet testin aikana

Ohjelmointivirheitä voi ilmentua, kun ohjelmaa lähdetään muuttamaan muille prosessoreille sopivaksi. Muistipaikkojen osoitemuuttujien word1...11 reunoilla (256 wordin välein) olevat "rikkinäiset" bitit saattavat olla ohjelmointivirheestä johtuvia. Syypäätä muistivirheeseen kannattaa etsiä bitnum-laskurin toimintaa ja alkusummaajaa tarkastelemalla.

Jos muistibitti ei ole lähelläkään osoitemuuttujia, on kyseessä todennäköisesti virhe muistissa, eli muisti on korruptoitunut. Muisti voi korruptoitua virtapiikkien ja jännitteenvaihteluiden seurauksena. Jos korruptoituneita muistipaikkoja ilmenee, täytyy testi-ohjelma ajaa laitteelle ja ohjelma toistaa useaan kertaan niin pitkään, että voidaan varmuudella todeta virhe väliaikaiseksi. Jos virhe toistuu, eikä se johdu ohjelmointivirheestä, ei prosessorin RAM-muisti ole tarpeeksi luotettava käyttöön tai ohjelmamuisti korruptoituu, kun se luetaan PROMista.

Yleensä prosessorit, joissa on viallinen muisti, eivät käynnisty. Jos viallisen laitteen kuitenkin saa päälle ja siihen jotenkin yhteyden, vika ilmentuu melko varmasti fault-tauluun. Jos laitteeseen saa yhteyden mutta se ei lähde run-tilaan, vika löytyy usein fault-taulusta. Toisinaan fault-taulu täytyy tyhjentää, koska joidenkin vikojen ilmetessä logiikka voi mennä suoraan stop-tilaan. Prosessorimoduulin asetuksista voidaan pakottaa ohjelma pysymään run-tilassa, vaikka virheitä ilmentuu.

8 Digitaaliset IO-moduulit

8.1 IC693MDR390 Mixed-IO -testi

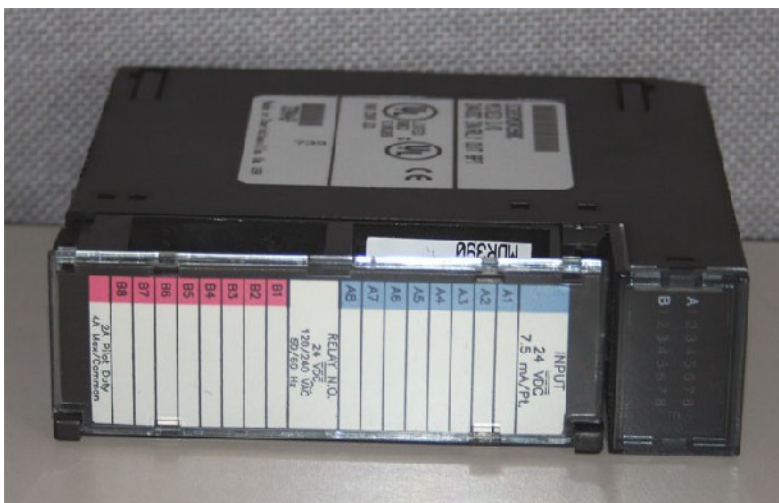
Digitaalisten IO-moduulien testauksessa jokaiselle moduulille täytyy rakentaa erilainen testausohjelma. Jokaista sisääntulomoduaalia varten pitää olla myös ulostulomoduaali. Tässä insinööriyössä perehdytäänkin IC693MDR390 Mixed-IO -moduulin testaukseen, koska siinä on saman verran sekä lähtöjä että tuloja. Tämän testin toimintaa voidaan käyttää perustana muiden digitaalisten IO-moduulien testaamiseen.

Testin tarkoitus on todeta IC693MDR390 mixed IO:n lähdöt ja tulot toimiviksi.

IC693MDR390-moduuli sisältää sisääntuloja ja ulostuloja saman määrän, joten sen lähtöjen ja tulojen testaus ei vaadi erillistä moduuliparia.

8.1.1 Laitteen kuvaus

IC693MDR390 sisältää 8 sisään- ja ulostuloa. Tulojen läpi kulkeva jännite tulee ulkoisesta virtalähteestä. Ulostulona voi käyttää myös 120V/240V AC:tä. Sisääntulona toimii 24V DC. Sisääntulot ovat optisesti eroteltuja.



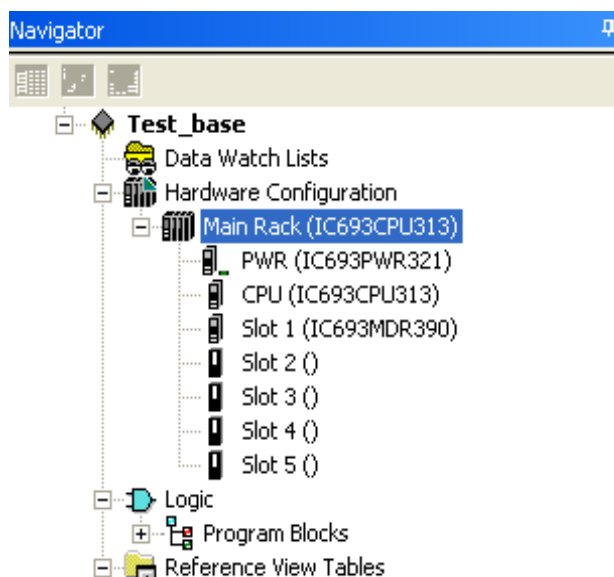
Kuva 20. IC693MDR390 Mixed-IO-moduuli

A1...A8 ovat sisääntuloja ja B1...B8 ovat ulostuloja. Päällä olevat ulos- ja sisääntulot näkyvät etupaneelissa olevista ledeillä valaistuista numeroista.

Sisään- ja ulostuloreleiden kestävyys riippuu käyttöjännitteestä. Alle 0,5 A virroilla releiden kesto aika on 800 000 operaatiota. Mitä korkeampi virta on, sitä lyhyempi elin aika on odotettavissa. 2 A virroilla odotettava elinikä on 200 000 operaatiota. [9]

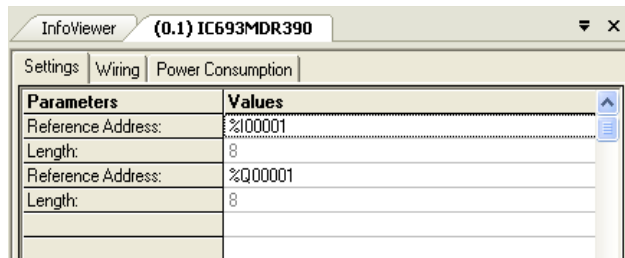
8.1.2 Hardware-asetukset

Moduuli on lisättävä ohjelmointiohjelmalla ohjelman hardware-asetuksiin, ennen kuin sitä voi käyttää. Lisääminen onnistuu PME:llä suoraan navigointipalkista. Valitaan oikea laitteisto, johon moduuli halutaan lisätä. Valitaan slot, mihin moduuli on kiinnitetty, ja valitaan listasta moduuli, joka on IC693MDR390.



Kuva 21. PME Navigator ja hardware asetukset

Kun moduuli on lisätty, täytyy moduulin tuloille ja lähdöille määrittää referenssiosoitteet. Referenssiosoitteilla varataan PLC:n muistista osiot lähdöille ja tuloille, joiden avulla IO:hon päästään käsiksi logiikkaohjelmalla. Referenssiosoitteita pääsee muokkaamaan Navigator-palkin hardware-osiosta painamalla hiiren oikeaa näppäintä moduulin kohdalla ja valitsemalla configure.

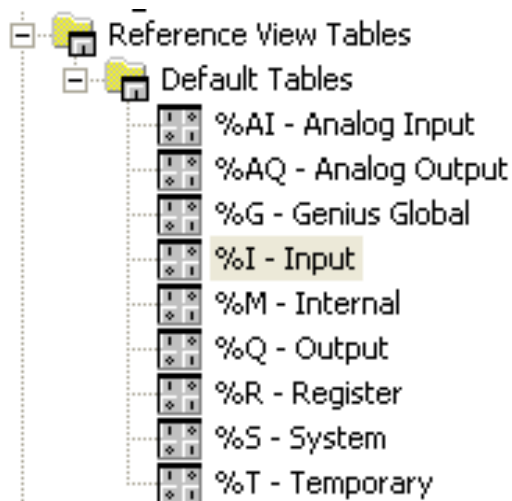


Kuva 22. PME mixed io:n konfiguraatio

Configure-ikkunasta IO-osoitteiksi asetetaan %I tuloille ja %Q lähdöille. Length-kohtiin merkitään, montako IO:ta moduulille halutaan varata. Tässä tapauksessa varataan 8, koska tuloja ja lähtöjä on kumpaakin 8.

Tässä vaiheessa sisään- ja ulostuloja voi jo testata käsin, jos ohjelman teko ei kiinnosta tai IO:ta on vähän. Sisääntulojen testaamista varten voi ottaa jännitettä virtalähteen 24 V jännitelähteestä. Ulostulojen testaamista varten riittää yleismittarin piipperi.

Muuttujien tiloja voi seurata ohjelmointiohjelmalla online-tilassa. Muuttujat löytyvät Navigator-ikkunasta reference view tables -> default tables. Sisääntuloja voi tarkastella %I:stä ja ulostuloja %Q:sta tai sieltä, mihin ne on itse asettanut kuvan 22 tavalla.

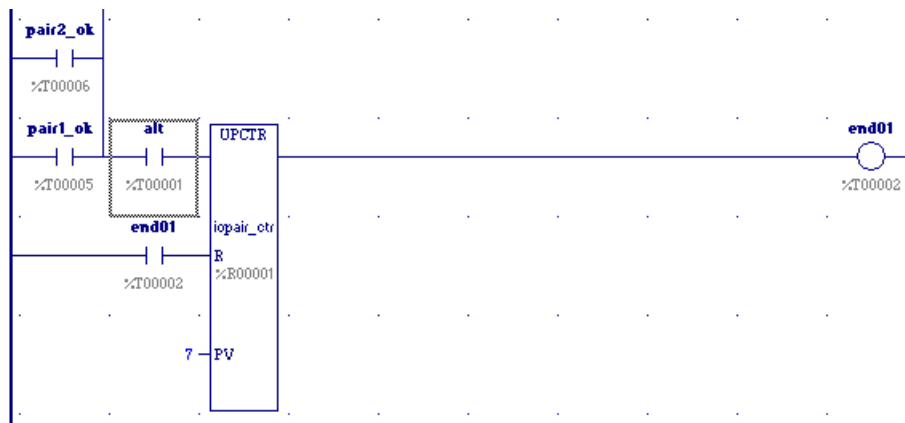


Kuva 23. Muistin bittitaulut

8.1.3 Testiohjelma

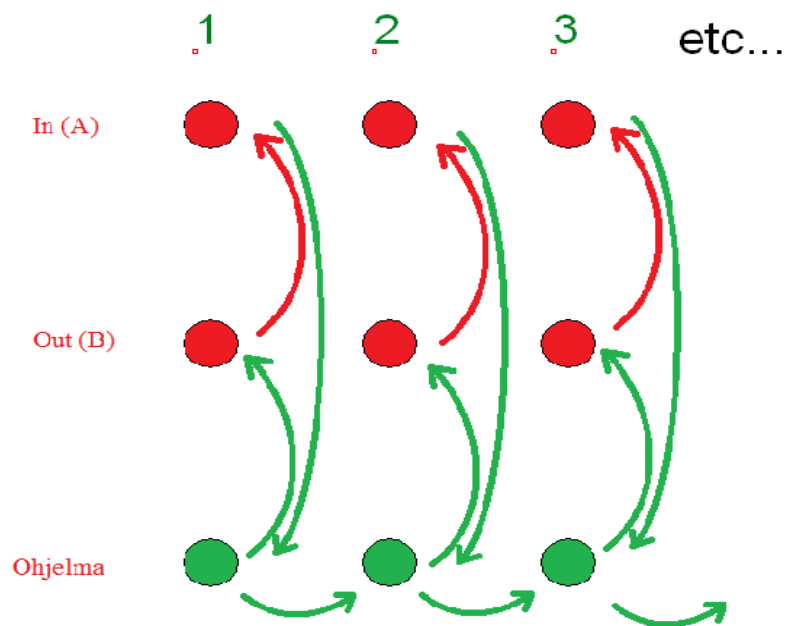
Testiohjelmassa käydään läpi yksi IO-parin kerrallaan. IO-pariin kuuluu yksi lähtö ja yksi ulostulo. Jokaiselle lähdölle ja tulolle luodaan oma muuttujansa prosessorin muistiin, jotta niihin voidaan viitata logiikkaohjelmassa. Jokaiselle parille luodaan OK-muuttuja. Ohjelman käynnistyessä ensimmäisen IO-parin numero on automaattisesti 0. IO-pari 0 tarkoittaa lähtöä B1 sekä sisääntuloa A1. IO-parissa 1 ovat B2 ja A2 ja niin edelleen.

Ohjelmaan luodaan IO-pair-laskuri, jonka arvo kasvaa yhdellä aina sitä mukaan, kuin IO-pari todetaan toimivaksi. Laskurissa on nopeussäätimenä alt-muuttuja, joka vaihtelee päälle ja pois halutulla nopeudella. IO-pair-laskuri määrää mitä IO-paria tarkastellaan milloinkin, joten IO-parien OK-muuttujat eivät voi olla päällä samanaikaisesti.



Kuva 24. IO-pair-laskuri. Parien 3-8 OK-muuttujat ovat jääneet pois kuvasta.

IO-parien OK-muuttujat ovat päällä, kun ohjelmassa parin lähtö %Q on asetettu päälle ja ohjelma lukee tulosta %I-lähdön antaman signaalin. Jos moduuli osaa lähettää ja lukea viestejä, voi sen hyvällä mielellä todeta toimivaksi. Kun IO-pari todetaan toimivaksi, kasvaa iopair-laskuri ja ohjelma siirtyy tarkastelemaan seuraavaa paria.



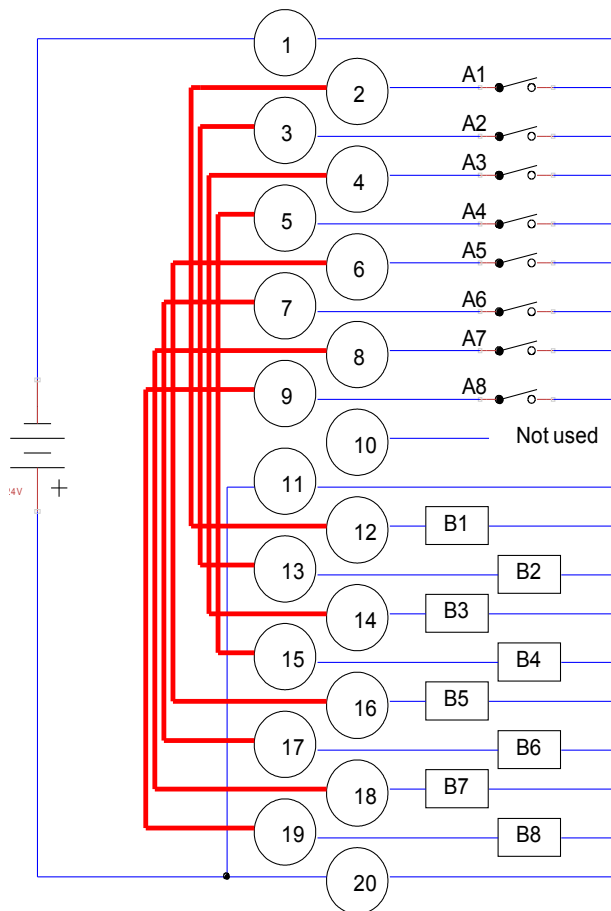
Kuva 25. Havainnollistus ohjelman etenemisestä. Vihreät nuolet ovat ohjelmallisia operaatioita. Punaiset ovat fyysisiä kytkentöjä.

Kuvassa 25 ohjelma asettaa Out(B):n päälle, josta lähtee johdinta pitkin jännite In(A):lle. Ohjelma lukee In(A):lta signaalin, jolloin ohjelmakierto on onnistunut ja IO-pari todetaan toimivaksi. Ohjelma siirtyy tämän jälkeen tarkastelemaan seuraavaa IO-paria ja niin edelleen.

8.1.4 Kytkenät

IC693MDR390-moduulin kytkennät tehdään suoraan moduulin ulostulosta sisääntuloihin. Moduuli ei itsessään anna jännitettä ulos, vaan ohjaa releitä, joten ulostuloon täytyy tuoda jännite ja sisääntulot täytyy maadoittaa. Jännitelähteenä voi toimia erillinen jännitelähde tai laitteiston oma virtalähde.

Testiohjelmaa varten tehdään seuraavat kytkennät logiikan virtalähteestä saatavalla 24V jännitteellä.



Kuva 26. Johdinkytännät testin ajaksi

8.1.5 Testaus

Testaus suoritettiin edellä mainittujen toimintatapojen mukaisesti. Logiikka kävi lävitse kaikki IO:t yksi kerrallaan, mikä oli selvästi nähtävissä IO-moduulin etupaneelissa olevista led-valoista. Led-valot ilmaisivat, mitkä IO:t ovat aktiivisia. Rikkinäistä IO:ta simuloitiin erillisellä kytkimellä, jolla saatiin yksi output -> input -yhteys katkaistua. Kytkimellä katkaisu aiheutti ohjelman kulun pysähtymisen "rikkinäisen" IO:n kohdalla. Näin voitiin todeta ja paikantaa rikkinäinen IO.

9 Päätelmät

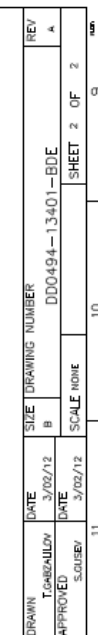
Nämä testiohjelmat testaavat moduulien toimintaa aivan sieltä alkukantaisimmasta päästä. Täysin 100 %:n varmuutta ne eivät anna kaikkiin mahdollisiin käyttökohteisiin, koska tietotekniikan kanssa voi ilmetä mitä tahansa odottamattomia ongelmia.

Nämä testit ovat verrattavissa auton toiminnan kokeiluun. Jos auton tankkiin voi laittaa bensiiniä, auto lähtee virta-avaimesta kääntämällä käyntiin ja sillä voi ajaa, voi auton todeta toimivaksi. Vaikka auto kulkee, ei se välttämättä menisi katsastuksesta läpi. Nämä moduulitestit ovat aivan yhtä yksinkertaisia. Monimutkaisia testejä on turha tehdä erikseen, koska GE Fanucin logiikoissa on jo jonkinlainen oma diagnostiikkansa. Näillä testeillä tarkistetaan logiikoiden perustoimivuus käytännössä (että niitä voi "ajaa"), vain ja ainoastaan sitä varten, että valmistajan tuotetuki loppuu, eikä korvaavaa laitetta ole saatavilla valmistajalta.

Lähteet

1. Navis Engineering Oy (www-sivusto)
<<http://www.navisincontrol.com/company/about-us.html>>. Luettu 30.3.2012
2. Programmable logic controller (www-sivusto),
<http://en.wikipedia.org/wiki/Programmable_logic_controller>.
Luettu 6.4.2012
3. Frank D. Petruzella. Programmable Logic Controllers. 3rd Edition. Mcgraw Hill.
(s. a.)
4. GE Intelligent Platforms (www-sivusto).
<http://www.ge-ip.com/products/2825>>.
Luettu 23.4.2012
5. 2007 GE FANUC 90/30 and 90/70 Serial Device Driver Guide (GE9030 &
GE9070) for devices capable of using SNP (Series Ninety Protocol). Version 6.0
Broadwin Technology, Inc. (pdf tiedosto).
<http://www.broadwin.com/downloads/pdf/GE_90_Serial.pdf>.
Luettu 15.4.2012
6. GE Fanuc CPU Instruction Set Reference Manual May 2002, GFK0467.
<http://support.ge-ip.com/support/resources/sites/GE_FANUC_SUPPORT/content/live/DOCUMENT/0/DO704/en_US/gfk0467m.pdf>.
Luettu 11.05.2012
7. GE Intelligent Platforms (PDF-tiedosto). Series 90*-30 PLC Installation and
Hardware Manual, GFK0356q.
<http://support.ge-ip.com/support/resources/sites/GE_FANUC_SUPPORT/content/staging/DOCUMENT/0/DO696/en_US/5.0/gfk0356q.pdf>.
Luettu 26.4.2012
8. GE Intelligen Platforms (pdf tiedosto). GFK1512R
<http://support.ge-ip.com/support/resources/sites/GE_FANUC_SUPPORT/content/live/DOCUMENT/0/DO662/en_US/GFK1512R.pdf>.
Luettu 10.4.2012
9. Series 90–30 PLC I/O Module Specifications – July 2000 (pdf tiedosto). GFK–
0898F.
<<http://www.pdfsupply.com/pdfs/gfk0898f.pdf>>.
Luettu 10.4.2012

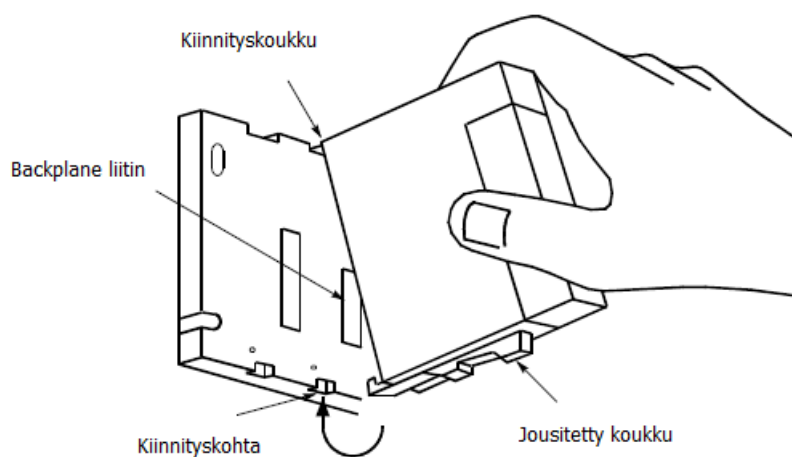
[illegible]



Moduulien kiinnitys- ja irrotusohje

Kiinnitys

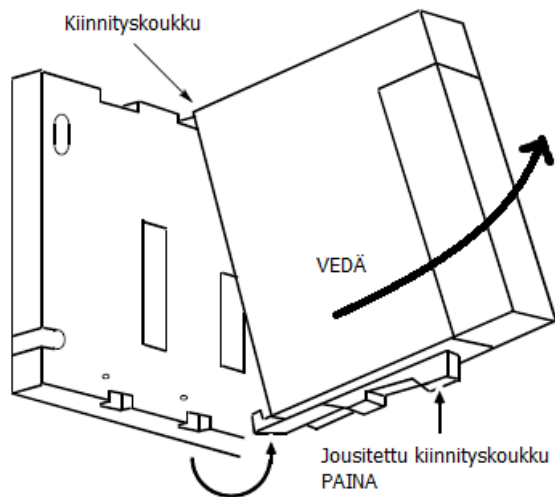
1. Irrota virtajohdot. Moduuleja ei saa lisätä jännitteisenä.
2. Tarkista, että moduulin sarjanumero on yhtenäinen hardware-konfiguraatiossa asetettuun moduulin sarjanumeroon. Jokaisella backplanen slotilla on omat moduuliasetukset (hardware-asetukset), joten moduuleita ei voi lyödä kiinni mihin sattuu. Virtalähde kiinnitetään vasemmalta ensimmäiseen kohtaan, sitten CPU- ja sitten IO-moduulit.
3. Moduulien takaseinässä on kiinnityskoukku. Aseta kiinnityskoukku backplanen reunalla olevaan kiinnityskohtaan, paina ja käännä moduulia alas ja kiinni backplaneen, kunnes se asettuu paikalleen niin, että moduulin alareunassa oleva jousikiinnitys napsahtaa paikalleen.
4. Tarkista, että moduuli on tiukasti kiinni.



Kuva 1: Moduulin kiinnitys backplaneen [7]

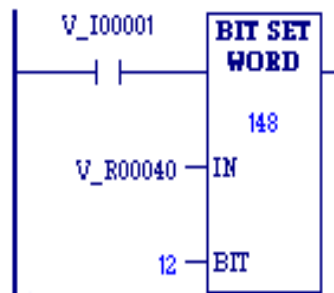
Irrotus

1. Irrota virta.
2. Paina jousitettua kiinnityskoukkuja niin, että se irtoaa kiinnikkeestä.
3. Vedä ja käännä moduulia samalla ylöspäin, kunnes se irtoaa.



Kuva 2: Moduulien irrotus backplanesta [7]

Esimerkki muistiin kirjoittamisesta BIT_SET_WORD-lohkolla.



Kuva 1: Bit_set_word lohko

Sisääntulot: IN Muuttuja: V_R00040 Osoite: %R00040
 BIT arvo: 12

Näillä tuloilla kahdestoista (12) bitti osoitteen %R00040 ensimmäisestä bitistä laskettuna vaihtuu todeksi.

%R40	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bitti	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit_set_word-lohkoilla voi kirjoittaa kuitenkin pidemmillekin pätkille, kuin vain yhteen word-alueeseen.

%R40:n ollessa osoitteena (IN) ja BIT-sisääntulon muuttujan arvon ollessa 24 (yli %R40 word-alueen) ollaan %R00041-paikan kahdeksannessa bitissä. BIT-sisääntuloa kasvattamalla saadaan muistia kirjoitettua yksi bitti kerrallaan.

%R40	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)
%R41	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	(17)	(18)	(19)	(20)	(21)	(22)	(23)	(24)	(25)	(26)	(27)	(28)	(29)	(30)	(31)	(32)

Numero 148 ilmoittaa monenko wordin päähän lohkoilla voi kirjoittaa. 148:n wordin alueella voi kirjoittaa %R40-%R188 muistipaikkojen bitteihin. Maksimikoko alueelle on 256 wordia.

System-tilatiedot

Osoite	Nimi	Kuvaus
%S0001	FST_SCN	Päällä, kun ohjelman kierros on ensimmäinen.
%S0002	LST_SCN	Resetoituu 1:stä 0:aan, kun ohjelmakierros on viimeinen.
%S0003	T_10MS	0.01 sekunnin ajastinkontakti.
%S0004	T_100MS	0.1 sekunnin ajastinkontakti.
%S0005	T_SEC	1.0 sekunnin ajastinkontakti.
%S0006	T_MIN	1.0 minuutin ajastinkontakti.
%S0007	ALW_ON	Aina päällä (1).
%S0008	ALW_OFF	Aina pois päältä (0).
%S0009	SY_FULL	Päällä, kun PLC fault -taulu on täynnä.
%S0010	IO_FULL	Päällä, kun IO fault -taulu on täynnä
%S0014	PLC_BAT	Päällä, kun virtalähteen patteri on tyhjä.
%SA0001	PB_SUM	Päällä, kun PLC havaitsee eroavaisuuksia muistin ja ohjelman välillä. Muisti korruptoitunut.
%SA0003	APL_FLT	Päällä, kun ohjelmallisia virheitä on havaittu.
%SA0009	CFG_MM	Päällä, kun PLC havaitsee eroavaisuuksia hardwaren ja konfiguraation välillä.
%SA0010	HRD_CPU	Päällä, kun diagnostiikka löytää virheitä CPU:sta.
%SA0011	LOW_BAT	Päällä, kun patterissa alhainen jännite.
%SA0014	LOS_IOM	Päällä, kun PLC menettää yhteyden IO-moduuliin.
%SA0015	LOS_SIO	Päällä, kun PLC menettää yhteyden varamoduuliin
%SB0010	BAD_RAM	Päällä, kun CPU havaitsee muistin korruptoituneen käynnistystyksen yhteydessä.
%SB0013	SFT_CPU	Päällä, kun CPU havaitsee virheen ohjelmassa.
%SC0009	ANY_FTL	Päällä, kun mikä tahansa virheilmoitus on aktiivinen.
%SC0010	SY_FLT	Päällä, kun mikä tahansa virheilmoitus on aktiivinen PLC virhetaulussa.
%SC0011	IO_FLT	Päällä, kun mikä tahansa virheilmoitus on aktiivinen IO virhetaulussa.
%SC0012	SY_PRES	Päällä, kun mikä tahansa virhe on luettavissa PLC virhetaulusta.
%SC0013	IO_PRES	Päällä, kun mikä tahansa virhe on luettavissa IO virhetaulusta.
%SC0014	HRD_FLT	Päällä, kun havaitaan virhe hardwaressa.
%SC0015	SFT_FLT	Päällä, kun havaitaan ohjelmallinen virhe.